UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/981,453	10/18/2001	Jens B. Junkermann	10022/093	6887

33391 7590 01/12/2007
BRINKS HOFER GILSON & LIONE
ONE INDIANA SQUARE, SUITE 1600
INDIANAPOLIS, IN 46204

EXAMINER	
STEVENS, ROBERT	
ART UNIT	**PAPER NUMBER**
2162	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	01/12/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No.	Applicant(s)	
	09/981,453	JUNKERMANN, JENS B.	
	Examiner	Art Unit	
	Robert Stevens	2162	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *29 September 2006*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *21-22,24-30,32,33,41-50,64-77 and 79* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *21-22,24-30,32-33,41-50,64-77 and 79* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. The Office withdraws the previous rejections of the claims under 35 USC §103(a), in light of the amendment. However, the Office sets forth new rejections of the claims under 35 USC §§101, 112-2nd paragraph and 103(a).

Response to Arguments

2. Applicant's arguments with respect to claims 21-22, 24-30, 32-33, 41-50, 64-77 and 79 have been considered but are moot in view of the new ground(s) of rejection.

Claim Objections

3. **Claims 21 and 41 are objected to** because of the following informalities:

Regarding claim 21: Lines 8-9 and 15-16 contains a grammatical error ("units ... is limited").

Regarding claim 41: Line 9 contains a grammatical error ("as wrapper").

Appropriate correction is required.

Claim Rejections - 35 USC § 101

4. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

5. **Claims 21-22, 24-30, 32-33, 41-50, 64-77 and 79 are rejected under 35 U.S.C. 101** because the claimed invention is directed to non-statutory subject matter.

To be statutory, a claimed computer-related process must either: (A) result in a physical transformation outside the computer for which a practical application is either disclosed in the specification or would have been known to a skilled artisan, or (B) be limited to a practical application with useful, concrete and tangible result.

A practical application can be either physical transformation or a useful, concrete and tangible result.

Regarding independent claims 21, 41 and 64: These claims are essentially directed to a series of data manipulations that take place within a computer. The claimed elements appear to be an aggregation of data into a structure rather than a practical application of that structure with a tangible result that enables any usefulness of the results to be realized. In other words, each of these claims describes the transformation of a data structure, however, there is no positively recited use of that data structure producing a tangible result.

Claims 21, 41, 64, and other claims that depend on them, are not patent eligible because the invention recited therein does not produce a useful, concrete and tangible result.

Claim Rejections - 35 USC § 112

6. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

7. **Claims 21-22, 24-30, 32-33, 41-50, 64-77 and 79 are rejected under 35 U.S.C. 112, second paragraph**, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Regarding claim 21: First, there is a lack of antecedent basis as to "the data structure" in lines 7 and 14. Second, it is unclear what is meant by the line 3 recitation of "developing ... code in a subclass". It is unclear how one "develops" code in a subclass. Third, there appears to be missing essential steps/elements. For instance, line 3 recites "developing" code, then line 5 recites "translating the request". It is unclear how one goes from developing code to translating requests. Additionally, a "request" was not positively recited as having been received (before translation takes place). Fourth, there are multiple references to "data" (lines 4, 8, 11, 13, 14, etc.). It is unclear whether each instance refers to the same or different "data" elements. Fifth, the limitation "while the data is read in" (line 14) is confusing. It is unclear if the intent is "while the data is being read in" or "after the data is read in". Sixth, the last limitation (lines 18-19) is directed to a translation process "based on the delivery technology". It is unclear what translation is taking place, what the end result is, and how "the delivery technology" is ascertained. It is noted that no "delivery technology" was ever determined/received/etc. There is also a lack of antecedent basis as to "the delivery technology" of line 19.

Claims 22, 24-30, 32-33 and 71-75 depend upon claim 21, and are therefore likewise rejected.

Regarding claim 41: First, the preamble recites "leveraging extensible markup language technology", however the body of the claim contains no reference to extensible markup language technology. Additionally, the preamble recites providing "an interface between ... back-end ... and front-end system layer[s]", however the body of the claim contains no reference to interfaces or front-end/back-end layers. As such, there appears to be missing essential steps/elements in the body of the claim. Second, there are no criteria as to how to determine the meaning of "direct" in line 5 ("direct the translation"). As such, it is unclear what the meaning of "direct" is. Third, there are no criteria as to how to determine the meaning of "restrict manipulation" and "standardize the content" in line 10. As such, it is unclear what the meaning of each of these phrases is. Fourth, it is unclear what particular structure/functionality each of the Message class and Field class (lines 9 and 14) provide, as they are recited as both providing the same functionality. Fifth, it is unclear what the phrase "limit a format of corresponding fields included in the input message" means. (I.e., "Limit" in what manner? What fields are "corresponding fields"? And, why would the format of an input message be "limited", rather than received as is and then translated to another format?) Sixth, there are no criteria as to how to determine the meaning of "direct" in line 18 ("direct the execution") and "as a function of" in line 19 ("as a function of the input message"). As such, it is unclear what the meaning of each of these phrases is.

Claims 42-50, 76-77 and 79 depend upon claim 41, and are therefore likewise rejected.

Regarding claim 64: First, the preamble recites a "framework to interface delivery technologies with data", however the body of the claim contains no references to interfacing or delivery technologies. As such, there appears to be missing essential steps/elements in the body of the claim. Second, There is a lack of antecedent basis as to "the data structure" recited in lines 8 and 15. Third, it is unclear what "data responsive to the request" (lines 11-12) is and what criteria are used to determine that data is "responsive".

Claims 65-70 depend upon claim 64, and are therefore likewise rejected.

Thus the scope of each of claims 21-22, 24-30, 32-33, 41-50, 64-77 and 79 is vague and ambiguous.

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

9. **Claims 21-22, 24-30, 32-33, 41-50, 64-77 and 79 are rejected under 35 U.S.C. 103(a)** as being unpatentable over Merrick et al. (US Patent No. 7,028312, filed Mar. 23, 1999 and issued Apr. 11, 2006, hereafter referred to as "Merrick") in view of Moore et al. (US Patent Application Publication No. 2002/0124045, filed Dec. 22, 2000 and published Sep. 5, 2002, hereafter referred to as "Moore").

Regarding independent claim 21: Merrick teaches ***A method of operating a business services application for retrieving data with delivery technologies, the method comprising: translating the request to a first document object model document with an ApiService class;*** (See Merrick col. 21 lines 25-32, discussing extraction of message arguments and use of the DOM.) ***during the translation, limiting the data structure of the first document object model document to representation as an input message with a plurality of fields, wherein units of data included in each of the fields is limited to a data type that is pre-specified in the business services application;*** (See Merrick col. 21 lines 10-14, discussing the use of labels found in the message/document.) ***executing the custom application code to retrieve data based on the first document object model document;*** (See Merrick col. 21 lines 10-14, discussing a programmer's use of the DOM API.) ***reading data into a second document object model document with the ApiService class;*** (See Merrick col. 21 lines 50-56, discussing the extraction of output arguments.) ***while the data is read in, limiting the data structure of the second document object model document to representation as an output message with a plurality of fields, wherein units of data included in each of the fields is limited to a data type that is pre-specified in the business services application;*** (See Merrick col. 21 lines 50-65 in the context of col. 21 lines 7-

14, teaching extraction of output arguments and use of the DOM API functionality.) ***and translating the second document object model document with the ApiService class based on the delivery technology.*** (See Merrick col. 21 lines 60-65, discussing encoding the output arguments.)

However, Merrick does not explicitly teach the further limitations as claimed. Moore, though, discloses ***developing custom application code in a subclass of a BusinessService class, the custom application code responsive to a request for data initiated by the delivery technologies;*** (See Moore paragraph [0058], discussing the use of business logic interfaces for messages.)

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Moore for the benefit of Merrick, because to do so allowed for a reduction in manpower required to make changes to a front-end system, as taught by Moore in paragraph [0003]. These references were all applicable to the same field of endeavor, i.e., XML message translations.

Regarding claim 22: Merrick does not explicitly teach the remaining limitations as claimed. Moore, though, discloses ***wherein limiting the data structure of the first document object model document comprises populating a plurality of text nodes within the first document object model document with request parameters contained in the request that are translated to a format identified with the pre-specified data type.*** (See Moore paragraph [0010], discussing the use of string nodes the use of string data types.)

Regarding claim 24: Merrick does not explicitly teach the remaining limitations as claimed. Moore, though, discloses ***wherein limiting the data structure of the first document object model document further comprises limiting the predetermined data type to a format of a string data type.*** (See Moore paragraph [0010], teaches the well known use of string data types.)

Regarding claim 25: Merrick does not explicitly teach the remaining limitations as claimed. Moore, though, discloses ***wherein limiting the data structure of the first document object model document comprises populating an at-tribute node within the first document object model document with an attribute of the request that is translated to a format identified with the pro-specified data type.*** (See Moore paragraph [0087], teaching the use of order5 attributes.)

Regarding claim 26: Merrick does not explicitly teach the remaining limitations as claimed. Moore, though, discloses ***further comprising selecting, as a function of a mode debug flag, to use one of a short field name or a long field name as a field name for each of the fields in the first and second document object model documents.*** (See Moore paragraph [0110], teaching code for a task name field and a task ID field.)

Regarding claim 27: Merrick does not explicitly teach the remaining limitations as claimed. Moore, though, discloses ***wherein the pre-specified data types are selected from a pre-specified group of data types consisting of a string data type, a long data type, an integer data type, a boolean data type and a group data type.*** (See Moore paragraph [0110], teaching the use of a string data type.)

Regarding claim 28: Merrick does not explicitly teach the remaining limitations as claimed. Moore, though, discloses ***wherein limiting the data structure of the first document object model document comprises loading a static declaration of a data type based on a list of fields expected in the request.*** (See Moore paragraph [0110], teaching the use of a public data type.)

Regarding claim 29: Merrick teaches ***wherein limiting the data structure of the second document object model document comprises populating a plurality of text nodes within the second document object model document with data read in to the second document object model document, wherein the format of the data that is read in is converted based on the data type.*** (See Merrick col. 22 lines 3-10, discussing the building of a reply message using "type information".)

Regarding claim 30: Merrick teaches ***wherein limiting the data structure of the second document object model document comprises populating an attribute node within the second document object model document with an attribute read in to the second document object***

model document that is translated to a format identified with the pre-specified data type. (See Merrick col. 22 lines 3-10, discussing the building of a reply message using "type information".)

Regarding claim 32: Merrick teaches ***wherein translating the second document object model comprises translating the second document object model document to extensible markup language text.*** (See Merrick col. 22 lines 3-10, discussing the building of a reply XML message using "type information".)

Regarding claim 33: Merrick teaches ***wherein translating the second document object model comprises translating the second document object model document to at least one of a hypertext markup language and a website meta language as a function of at least one extensible stylesheet language stylesheet.*** (See Merrick col. 22 lines 3-10, discussing the building of a reply XML message using "type information".)

Regarding claim 71: Merrick teaches ***wherein limiting the data structure of the first document object model comprises standardizing the format of the document object model to be substantially similar for a similar request received from any one of the delivery technologies.*** (See Merrick col. 21 lines 10-15, discussing the use of the DOM in the implementation of a generic encoding structure.)

Regarding claim 72: Merrick teaches ***wherein limiting the data structure of the second document object model comprises standardizing the format of the second document object model to be compatible with any one of the delivery technologies.*** (See Merrick col. 21 lines 5-14, teaching the generation of a generic encoding structure for an RPC delivery technology mechanism.)

Regarding claim 73: Merrick teaches ***wherein executing the custom application code comprises executing the same custom application code for a similar request from any one of the delivery technologies to provide a response.*** (See Merrick col. 21 lines 5-14, teaching the generation of a generic encoding structure for an RPC delivery technology mechanism.)

Claim 74 is substantially similar to claim 73, and therefore likewise rejected.

Regarding claim 75: Merrick teaches ***wherein while the data is read in, limiting the data structure of the second document object model document comprises similarly limiting the second document object model in response to similar requests from any of the delivery technologies.*** (See Merrick col. 21 lines 5-14, teaching the generation of a generic encoding structure for an RPC delivery technology mechanism.)

Regarding independent claim 41: Merrick teaches ***A system for leveraging extensible markup language technology to provide an interface between a back-end systems layer and a front-end systems layer, the system comprising: a server computer;*** (See Merrick Fig. 1, showing a "Server Machine".) ***an ApiService class operable within the server computer to direct the translation of a request to an input message that includes a plurality of fields;*** (See Merrick col. 21 lines 7-14, discussing the use of the DOM API for message conversions.) ***a document object model class operable within the server computer to represent the input message as a document object model document;*** (See Merrick col. 21 lines 39-44 in context of col. 21 lines 10-14, teaching the use of a DOM API to represent an input message.) ***a Message class and a Field class operable within the server computer as wrapper of the document object model class to restrict manipulation and standardize the content of the document object model document;*** (See Merrick col. 21 lines 26-31, discussing the use of wrappers to encapsulate a message representation.)

However, Merrick does not explicitly teach the further limitations as claimed. Moore, though, discloses ***a MESSAGEDEFINITION class operable in the server, wherein the MESSAGEDEFINITION class includes a listing of pre-specified fields each of which describe a corresponding pre-specified data type, and wherein the Message class and the Field class are further operable within the server during translation to limit a format of corresponding fields included in the input message to a predetermined data structure based on the described corresponding pre-specified data type;*** (See Moore paragraph [0081] in the context of paragraphs [0087] and [0090], teaching the use of MessageData objects [i.e., instantiated classes] for the processing of message arguments or fields.) ***and a BusinessService class operable within***

the server computer to direct the execution of custom application code as a function of the input message, wherein the custom application code includes a pro-specified data type to limit the format of those fields included in the input message that do not correspond to the listing of pre-specified fields. (See Moore paragraph [0058], discussing the use of business logic interfaces for messages.)

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Moore for the benefit of Merrick, because to do so allowed for a reduction in manpower required to make changes to a front-end system, as taught by Moore in paragraph [0003]. These references were all applicable to the same field of endeavor, i.e., XML message translations.

Regarding claim 42: Merrick does not explicitly teach the remaining limitations as claimed. Moore, though, discloses ***wherein the custom application code is operable to process the input message to retrieve data, the data translatable with the document object model class, the Message class and the Field class to an output message in the form of a document object model document with restricted manipulation and standardized content based on the pre-specified data type included in the custom application code that, during translation, is operable to limit a format of each of a plurality of fields included in the output message to a predetermined data structure.*** (See Moore paragraph [0081] in the context of paragraphs [0087] and [0090], teaching the use of MessageData objects [i.e., instantiated classes] for the processing of message arguments or fields.)

Regarding claim 43: Merrick teaches ***wherein the ApiService class is operable to direct the conversion of the output message to a presentation format defined by the request.*** (See Merrick col. 22 lines 3-10 in context of col. 21 lines 46-49, discussing the building of a reply message in accordance with "type information".)

Regarding claim 44: Merrick does not explicitly teach the remaining limitations as claimed. Moore, though, discloses ***wherein the input message and the output message comprises a root element mid a plurality of sub-elements.*** (See Moore paragraph [0087], discussing the use of NodeData entries – it being well-known that DOM processing utilizes a tree data structure having a root node.)

Regarding claim 45: Merrick teaches ***further comprising a Fldtypes class operable within the server computer, wherein the Fldtypes class comprises definitions of the format of data types for fields within the input message.*** (See Merrick col. 22 lines 3-10, discussing the use of input message "type information" to build a reply message.)

Regarding claim 46: Merrick does not explicitly teach the remaining limitations as claimed. Moore, though, discloses ***wherein the document object model document comprises a plurality of field names, the field names selectable with a mode debug flag as one of a short field name and a long field name.*** (See Moore paragraph [0100] teaching error processing/debugging via exception handling, and paragraph [0110] teaching the sue of long [task name strings] and short [task ID] names.)

Regarding claim 47: Merrick does not explicitly teach the remaining limitations as claimed. Moore, though, discloses ***wherein the short field name and the long field name are defined in the MESSAGEDEFINITION class operable within the server computer.*** (See Moore paragraph [0087] discussing MessageData class/object structures, in the context of paragraph [0110] teaching the use of long [task name strings] and short [task ID] names.)

Regarding claim 48: Merrick teaches ***wherein the document object model class comprises a Document class, a document object model Element class and a plurality of ProcessingInstruction classes, the Message class operable as a wrapper of the Document class, the document object model Element class and the ProcessingInstruction classes.*** (See Merrick col. 21 lines 30-31, discussing the use of a wrapper to encapsulate.)

Regarding claim 49: Merrick teaches ***wherein the document object model class comprises a document object model setAttribute method, Field class operable as a wrapper of the document object model setAttribute method.*** (See Merrick col. 21 lines 30-31, discussing the use of a wrapper to encapsulate.)

Regarding claim 50: Merrick does not explicitly teach the remaining limitations as claimed. Moore, though, discloses ***wherein the BusinessService class comprises a subclass of custom application code responsive to the request.*** (See Moore paragraph [0058], discussing the use of business logic interfaces for messages.)

Regarding claim 76: Merrick does not explicitly teach the remaining limitations as claimed. Moore, though, discloses ***wherein the Message class and the Field class are operable during representation of the input message as the document object model document to restrict manipulation of the document object model document.*** (See Moore paragraph [0087], describing the use of a MessageData class having NodeData entries.)

Regarding claim 77: Merrick does not explicitly teach the remaining limitations as claimed. Moore, though, discloses ***wherein the Message class is operable to restrict creation of the clement nodes and population of the corresponding text nodes and the Field class is operable to restrict the data types of text and attribute nodes included in the first document object model document.*** (See Moore paragraph [0087], describing the use of a MessageData class having NodeData entries.)

Regarding claim 79: Merrick does not explicitly teach the remaining limitations as claimed. Moore, though, discloses ***wherein the pre-specified data types are selected from the group consisting of integer, long, Boolean, string and group.*** (See Moore paragraph [0110], describing the use of a string data type.)

Regarding independent claim 64: Merrick teaches ***An e-commerce architecture for providing a framework to interface delivery technologies with data, the e-commerce architecture comprising: a server computer operable to execute instructions to convert a request to a first document object model document in an extensible markup language, the first document object model document comprising a plurality of request parameters extracted from the request;*** (See Merrick col. 21 lines 39-42 discussing the extraction of arguments from an XML input message, in the context of col. 21 lines 7-14 discussing DOM processing.) ***the server computer operable to execute instructions to retrieve data responsive to the request and convert the data to a second document object model document in the extensible markup language based on the request parameters;*** (See Merrick col. 21 lines 50-55 discussing the creation of an XML message containing output arguments, in the context of col. 21 lines 7-14 discussing DOM processing.)

However, Merrick does not explicitly teach the further limitations as claimed. Moore, though, discloses ***the server computer operable to execute instructions to restrict the conversion to the first document object model document based on a listing of data types that are pre-specified for the request parameters, wherein the data types limit the data structure of a plurality of fields included in the first document object model document to a predetermined data structure specified by the data types;*** (See Moore paragraph [0081] in the context of paragraphs [0087] and [0090], teaching the use of MessageData objects [i.e., instantiated classes] for the processing of message arguments or fields.) ***and the server computer operable to execute instructions to restrict the conversion of the data to the second document object model document to limit the data structure of a plurality of fields included in the second document***

object model document to a predetermined data structure specified by the data types. (See Moore paragraph [0081] in the context of paragraphs [0087] and [0090], teaching the use of MessageData objects [i.e., instantiated classes] for the processing of message arguments or fields.)

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Moore for the benefit of Merrick, because to do so allowed for a reduction in manpower required to make changes to a front-end system, as taught by Moore in paragraph [0003]. These references were all applicable to the same field of endeavor, i.e., XML message translations.

Regarding claim 65: Merrick does not explicitly teach the remaining limitations as claimed. Moore, though, discloses ***wherein the instructions to restrict the conversion of the first and second document object model documents further comprise instructions executable by the server computer to identify the first and second document object model documents with a predefined name included in the request.*** (See Moore paragraph [0094] and the ensuing paragraphs, such as [0110], teaching the use of predefined node names.)

Regarding claim 66: Merrick teaches ***wherein the instructions to restrict the conversion of the first and second document object model documents further comprise instructions executable by the server computer to create a plurality of element nodes and populate a plurality of corresponding text nodes with the respective request parameters and***

the respective data. (See Merrick col. 21 lines 7-14, discussing DOM processing and the building of data structures identified by labels/filed names.)

Regarding claim 67: Merrick teaches ***wherein the instructions to restrict the conversion of the first and second document object model documents further comprise instructions executable by the server computer to define the data type of each of the text nodes from among a predefined group of data types***. (See Merrick col. 22 lines 3-10, describing the building of a message from "type information".)

Regarding claim 68: Merrick teaches ***wherein the instructions to restrict the conversion comprises a Message class operable as a wrapper of a plurality of classes within the document object model class that include a document class and a document object model element class***. (See Merrick col. 21 lines 30-31, describing the practice of encapsulation/wrapping.)

Regarding claim 69: Merrick teaches ***wherein the instructions to restrict the conversion comprises a Field class operable as a wrapper of a document object model setAttribute method in a document object model element class***. (See Merrick col. 21 lines 30-31, describing the practice of encapsulation/wrapping.)

Regarding claim 70: Merrick does not explicitly teach the remaining limitations as claimed. Moore, though, discloses ***wherein the instructions to retrieve data responsive to the request are identified with a request name that is included in the request.*** (See Moore paragraph [0110], teaching the use of a task name string and an ID.)

Conclusion

10. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Non-Patent Literature

Miyazawa, Tatsuo, et al., "An Advanced Internet XML/EDI Model Based on Secure XML Documents", Seventh International Conference on Parallel and Distributed Systems 2000, Iwate, Japan, Jul. 4-7, 2000, pp. 295-300.

US Patents

Ankireddipally et al	6,971,096
Lee et al	6,336,137
Lewallen	6,957,439

Contact Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Robert Stevens whose telephone number is (571) 272-4102. The examiner can normally be reached on M-F 6:00 - 2:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John E. Breene can be reached on (571) 272-4107. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Robert Stevens
Examiner
Art Unit 2162

January 4, 2007

MOHAMMAD ALI
PRIMARY EXAMINER